Отчет по лабораторной работе №2

Задача о погоне - вариант 30

Тимур Еюбоглу

Содержание

1 Цель работы **4**

2 Задание **5**

3 Выполнение лабораторной работы **6**
3.1 Условие задачи 8
3.2 Код программы (Julia) 8
3.3 Код программы (Python) 10
3.4 Решение 13

4 Выводы **16**

List of Figures

3.1 траектории для случая 1 (Python) 13
3.2 траектории для случая 1 (Julia) 13
3.3 траектории для случая 2 (Python) 14
3.4 траектории для случая 2 (Julia) 15

1 Цель работы

Приведем один из примеров построения математических моделей для выбора правильной стратегии при решении задач поиска. Например, рассмотрим задачу преследования браконьеров береговой охраной. На море в тумане катер береговой охраны преследует лодку браконьеров. Через определенный промежуток времени туман рассеивается, и лодка обнаруживается на расстоянии k км от катера. Затем лодка снова скрывается в тумане и уходит прямолинейно в неизвестном направлении. Известно, что скорость катера в n раза больше скорости браконьерской лодки. Необходимо определить по какой траектории необходимо двигаться катеру, чтоб нагнать лодку.

2 Задание

1. Провести необходимые рассуждения и вывод дифференциальных уравнений, если скорость катера больше скорости лодки в n раз.
2. Построить траекторию движения катера и лодки для двух случаев.
3. Определить по графику точку пересечения катера и лодки.

3 Выполнение лабораторной работы

Принимаем за $t_0 = 0, X_0 = 0$ - место нахождения лодки браконьеров в момент обнаружения, $X_0 = k$ - место нахождения катера береговой охраны относительно лодки браконьеров в момент обнаружения лодки.

Введем полярные координаты. Считаем, что полюс - это точка обнаружения лодки браконьеров $x_0 = 0(\theta = x_0 = 0)$, а полярная ось r проходит через точку нахождения катера береговой охраны.

Чтобы найти расстояние x (расстояние после которого катер начнет двигаться вокруг полюса), необходимо составить простое уравнение. Пусть через время t катер и лодка окажутся на одном расстоянии x от полюса. За это время лодка пройдет x, а катер $x - k$ (или $x + k$, в зависимости от начального положения катера относительно полюса). Время, за которое они пройдут это расстояние, вычисляется как $\frac{x}{v}$ или $\frac{x+k}{v}$ (для второго случая $\frac{x-k}{v}$). Так как время одно и то же, то эти величины одинаковы. Тогда неизвестное расстояние можно найти из следующего уравнения: $\frac{x}{v} = \frac{x+k}{v}$ - в первом случае, $\frac{x}{v} = \frac{x-k}{v}$ во втором случае.

Отсюда мы найдем два значения x_1 и x_2, задачу будем решать для двух случаев.

$x_1 = \frac{k}{n+1}$,при $\theta = 0$

$x_2 = \frac{k}{n-1}$,при $\theta = -\pi$

После того, как катер береговой охраны окажется на одном расстоянии от полюса, что и лодка, он должен сменить прямолинейную траекторию и начать двигаться вокруг полюса удаляясь от него со скоростью лодки v. Для этого скорость катера раскладываем на две составляющие: v_r - радиальная скорость и v_t- тангенциальная скорость. Радиальная скорость - это скорость, с которой катер удаляется от полюса $\upsilon_r = \frac{dr}{dt}$. Нам нужно, чтобы эта скорость была равна скорости лодки, поэтому полагаем $\upsilon = \frac{dr}{dt}$. Тангенциальная скорость – это линейная скорость вращения катера относительно полюса. Она равна произведению угловой скорости $\frac{d\theta}{dt}$ на радиус r, $\upsilon r = r\frac{d\theta}{dt}$ Найдем тангенциальную скорость для нашей задачи $\upsilon_t = r\frac{d\theta}{dt}$. Вектора образуют прямоугольный треугольник, откуда по теореме Пифагора можно найти тангенциальную скорость $\upsilon_t = \sqrt{n^2\upsilon_r^2 - \upsilon^2}$. Поскольку, радиальная скорость равна υ, то тангенциальную скорость находим из уравнения $\upsilon_t = \sqrt{n^2\upsilon^2 - \upsilon^2}$. Следовательно, $\upsilon_\tau = \upsilon\sqrt{n^2 - 1}$.

Тогда получаем $r\frac{d\theta}{dt} = \upsilon\sqrt{n^2 - 1}$

Решение исходной задачи сводится к решению системы из двух дифференциальных уравнений

$$\begin{cases} \frac{dr}{dt} = \upsilon \\ r\frac{d\theta}{dt} = \upsilon\sqrt{n^2 - 1} \end{cases}$$

с начальными условиями

$$\begin{cases} \theta_0 = 0 \\ r_0 = \frac{k}{n+1} \end{cases}$$

$$\begin{cases} \theta_0 = -\pi \\ r_0 = \frac{k}{n-1} \end{cases}$$

Исключая из полученной системы производную по t, можно перейти к следующему уравнению: $\frac{dr}{d\theta} = \frac{r}{\sqrt{n^2-1}}$

Начальные условия остаются прежними. Решив это уравнение, мы получим траекторию движения катера в полярных координатах. Теперь, когда нам известно все, что нам нужно, построим траекторию движения катера и лодки для двух случаев.

3.1 Условие задачи

На море в тумане катер береговой охраны преследует лодку браконьеров. Через определенный промежуток времени туман рассеивается, и лодка обнаруживается на расстоянии 12.2 км от катера. Затем лодка снова скрывается в тумане и уходит прямолинейно в неизвестном направлении. Известно, что скорость катера в 4.1 раза больше скорости браконьерской лодки

3.2 Код программы (Julia)

```
using DifferentialEquations
using Plots

n = 4.1
s = 12.2
fi = 3/4*pi

function f(r, p, t)
    dr = r/sqrt(n^2-1)
    return dr
end

function f2(t)
    xt = tan(fi+pi)*t
    return xt
end

r0 = s/(n+1)
```

```
theta0 = collect(LinRange(0, 2*pi, 10000))
prob = ODEProblem(f, r0, (0, 2*pi))
sol = solve(prob, saveat=theta0)

t = collect(LinRange(0.00000001, 8, 1000))
r1=[]
theta1=[]
for i in t
    push!(r1, sqrt(i^2 + f2(i)^2))
    push!(theta1, atan(f2(i)/i))
end

plot(sol, proj=:polar, label="катер")
plot!(theta1, r1, proj=:polar, label="лодка")

savefig("01jl.png")

r0 = s/(n-1)

theta0 = collect(LinRange(0, 2*pi, 10000))
prob = ODEProblem(f, r0, (0, 2*pi))
sol = solve(prob, saveat=theta0)

t = collect(LinRange(0.00000001, 17, 1000))
r1=[]
theta1=[]
for i in t
    push!(r1, sqrt(i^2 + f2(i)^2))
    push!(theta1, atan(f2(i)/i))
```

```
end

plot(sol, proj=:polar, label="катер")
plot!(theta1, r1, proj=:polar, label="лодка")

savefig("02jl.png")
```

3.3 Код программы (Python)

```
from math import *
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plot

n = 4.1
s = 12.2
fi = 3/4*pi

def f(theta, r):
    dr = r/sqrt(n**2 - 1)
    return dr

def f2(t):
    xt = tan(fi+pi)*t
    return xt

r0 = s/(n+1)

theta = np.arange(0, 2*pi, 0.01)
```

```
r = odeint(f, r0, theta)

t = np.arange(0.000000001, 20)
r1 = np.sqrt(t**2 + f2(t)**2)
theta1 = np.arctan(f2(t)/t)

plot.rcParams["figure.figsize"] = (10, 10)

plot.polar(theta, r, 'red', label = 'катер')
plot.polar(theta1, r1, 'green', label = 'лодка')

tmp = 0
for i in range(len(theta)):
    if round(theta[i], 2) == round(fi+pi, 2):
        tmp = i
print('Тета:', theta[tmp], "r:", r[tmp][0])

plot.legend()
plot.savefig("01.png",dpi=100)

r0 = s/(n-1)

theta = np.arange(0, 2*pi, 0.01)
r = odeint(f, r0, theta)

t = np.arange(0.000000001, 20)
r1 = np.sqrt(t**2 + f2(t)**2)
theta1 = np.arctan(f2(t)/t)
```

```
plot.rcParams["figure.figsize"] = (10, 10)

plot.polar(theta, r, 'red', label = 'катер')
plot.polar(theta1, r1, 'green', label = 'лодка')

tmp = 0
for i in range(len(theta)):
    if round(theta[i], 2) == round(fi+pi, 2):
        tmp = i
print('Тета:', theta[tmp], "r:", r[tmp][0])

plot.legend()
plot.savefig("02.png",dpi=100)
```

3.4 Решение

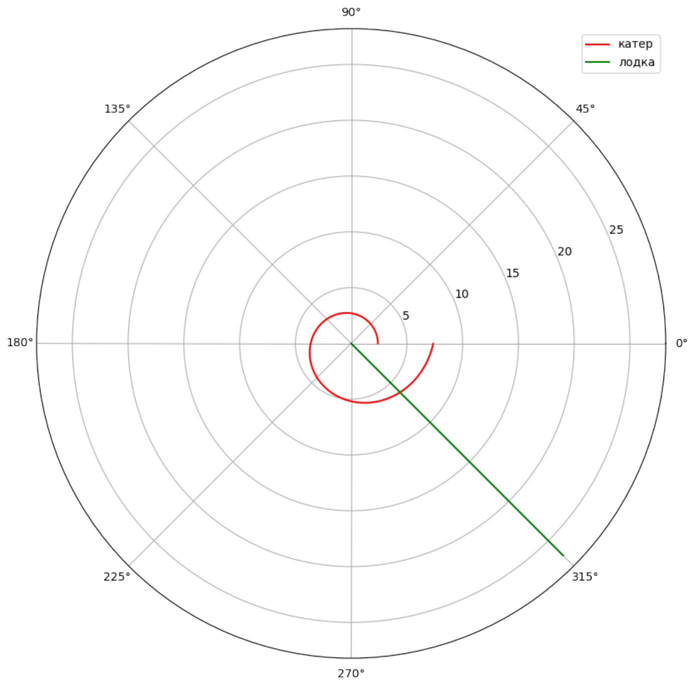

Figure 3.1: траектории для случая 1 (Python)

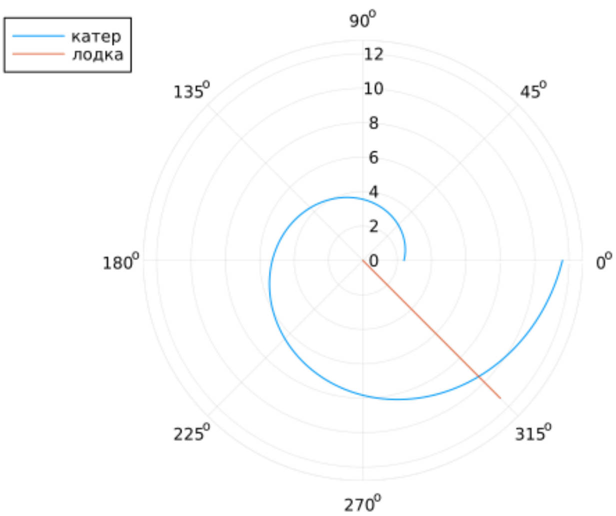

Figure 3.2: траектории для случая 1 (Julia)

Точка пересечения красного и зеленого графиков - точка пересечения катера и лодки, исходя из графика, имеет координаты

$$\begin{cases} \theta = 315 \\ r = 6.196 \end{cases}$$

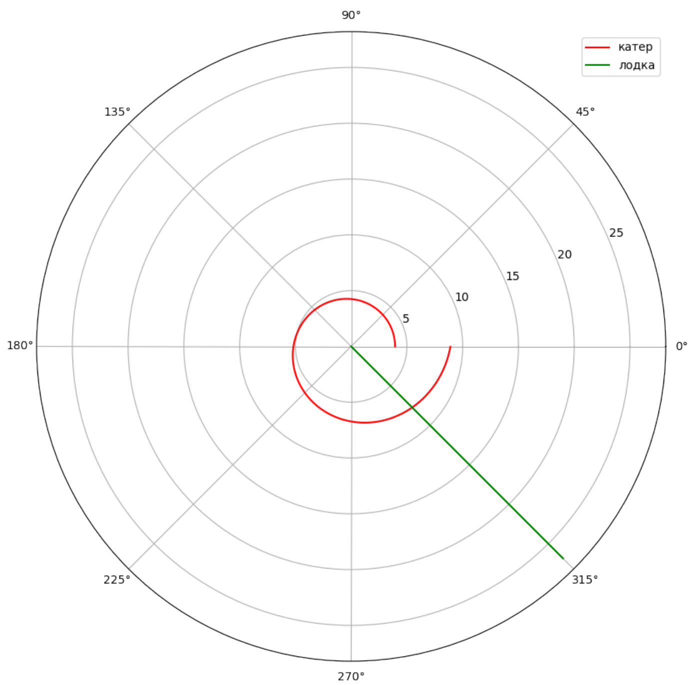

Figure 3.3: траектории для случая 2 (Python)

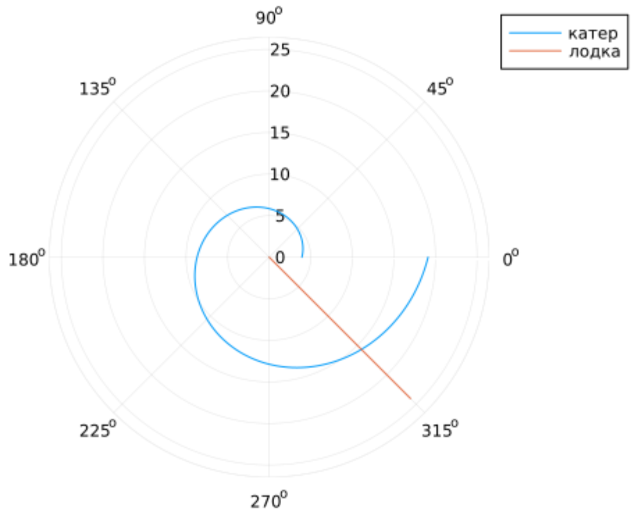

Figure 3.4: траектории для случая 2 (Julia)

Точка пересечения красного и зеленого графиков - точка пересечения катера и лодки, исходя из графика, имеет координаты

$$\begin{cases} \theta = 315 \\ r = 7.739 \end{cases}$$

Наблюдаем, что при погоне «по часовой стрелке» для достижения цели потребуется пройти меньшее расстояние.

4 Выводы

Рассмотрели задачу о погоне. Провели анализ и вывод дифференциальных уравнений. Смоделировали ситуацию.